

```
# Step 1: Install the required libraries
# Run `pip install transformers torch` if not already installed
```


```
# Step 2: Import the pipeline function
from transformers import pipeline
```

```
# Step 3: Initialize a question-answering pipeline
qa_pipeline = pipeline("question-answering")
```

```
# Step 4: Define the context and the question
context = """
Charles Babbage is often referred to as the "father of the computer"
because he conceptualized and invented the first mechanical computer in the early 19th century.
"""
question = "Who is known as the father of the computer?"
```

```
# Step 5: Use the pipeline to answer the question
result = qa_pipeline(question=question, context=context)
```

```
# Step 6: Display the result
print("Answer:", result['answer'])
print("Score:", result['score'])
print("Start index:", result['start'])
print("End index:", result['end'])
```

 No model was supplied, defaulted to distilbert/distilbert-base-cased-distilled-squad and revision 564e9b5 (<https://huggingface.co/distilbert/distilbert-base-cased-distilled-squad>)
Using a pipeline without specifying a model name and revision in production is not recommended.
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (<https://huggingface.co/settings/tokens>), set it as secret in your Colab secrets.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.

config.json: 100%	473/473 [00:00<00:00, 9.00kB/s]
model.safetensors: 100%	261M/261M [00:02<00:00, 136MB/s]
tokenizer_config.json: 100%	49.0/49.0 [00:00<00:00, 3.70kB/s]
vocab.txt: 100%	213k/213k [00:00<00:00, 3.63MB/s]
tokenizer.json: 100%	436k/436k [00:00<00:00, 9.54MB/s]

Device set to use cpu
Answer: Charles Babbage
Score: 0.9988206624984741
Start index: 1
End index: 16


```
# Step 1: Import the pipeline function
from transformers import pipeline
```

```
# Step 2: Initialize the question-answering pipeline with a custom model
qa_pipeline = pipeline(
    "question-answering",
    model="deepset/roberta-base-squad2"
)
```

```
# Step 3: Define the context and question
context = """
Charles Babbage is often referred to as the "father of the computer"
because he conceptualized and invented the first mechanical computer in the early 19th century.
"""
question = "Who is known as the father of the computer?"
```

```
# Step 4: Use the pipeline to answer the question
result = qa_pipeline(question=question, context=context)
```

```
# Step 5: Display the result
print("Answer:", result['answer'])
print("Score:", result['score'])
print("Start index:", result['start'])
print("End index:", result['end'])
```

 config.json: 100%	571/571 [00:00<00:00, 7.78kB/s]
model.safetensors: 100%	496M/496M [00:06<00:00, 165MB/s]
tokenizer_config.json: 100%	79.0/79.0 [00:00<00:00, 5.11kB/s]
vocab.json: 100%	899k/899k [00:00<00:00, 9.82MB/s]
merges.txt: 100%	456k/456k [00:00<00:00, 10.6MB/s]
special_tokens_map.json: 100%	772/772 [00:00<00:00, 33.9kB/s]
Device set to use cpu Answer: Charles Babbage Score: 0.950535237789154 Start index: 1 End index: 16	

```
# Step 1: Import the pipeline function
from transformers import pipeline
```

```
# Step 2: Initialize the question-answering pipeline with the custom model
qa_pipeline = pipeline(
    "question-answering",
    model="deepset/roberta-base-squad2"
)
```


```
# Step 3: Define a custom context
context = """
The Eiffel Tower is one of the most iconic landmarks in the world. Located in Paris, France,
it was completed in 1889 and stands at a height of 330 meters. The tower attracts millions of
visitors each year and is especially popular during the evening when it is illuminated.
"""
```

```
# Step 4: Define two questions
question_1 = "Where is the Eiffel Tower located?"
question_2 = "When was the Eiffel Tower completed?"
```

```
# Step 5: Use the pipeline to answer the questions
result_1 = qa_pipeline(question=question_1, context=context)
result_2 = qa_pipeline(question=question_2, context=context)
```

```
# Step 6: Display the results
print("Question 1:", question_1)
print("Answer 1:", result_1['answer'])
print("Score 1:", result_1['score'])
print("Start index 1:", result_1['start'])
print("End index 1:", result_1['end'])

print("\nQuestion 2:", question_2)
print("Answer 2:", result_2['answer'])
print("Score 2:", result_2['score'])
print("Start index 2:", result_2['start'])
print("End index 2:", result_2['end'])
```

```
 Device set to use cpu
Question 1: Where is the Eiffel Tower located?
Answer 1: Paris, France
Score 1: 0.928819477558136
Start index 1: 79
End index 1: 92

Question 2: When was the Eiffel Tower completed?
Answer 2: 1889
Score 2: 0.9824663400650024
Start index 2: 115
End index 2: 119
```

